Software Engineering

The process of building large and complex software systems in a cost-effective way."

In 1977, annual software costs in the United States were in excess of $50 billion (3% of the GNP).

It is estimated that these costs have now exceeded $800 billion.
The Five Phases of Software Development

Requirements
Design
Implementation
Testing
Maintenance
Requirements Phase

Define exactly what the user wants

- Appearance
- Performance
- Time Table

Discussion with User

- User provides his ideas on what he wants
- SE has to ask questions and clarify what the user has indicated
- SE must "fill in the gaps" left by the user's discussion with further questions
- SE must clarify the user's goals
- SE must talk at the user's level
Requirements Phase

SE produces a "Feasibility Study"

A document that describes any problems that may be encountered due to limitations based on the hardware/software facilities available

SE produces a "Software Requirements Document"

An SRD is a formal document that describes in detail exactly what the product is supposed to do (not necessarily how it does it)
Design Phase

Hierarchical Structure
(leads to Top-Down Design Specifications)

Data Structure Design

Pseudocode
(based on Hierarchy Chart and Data Structures)
Implementation Phase

Pseudocode and Data Structures are transformed into code

Top-Down Implementation Approach

Drivers and Stubs introduced

Testing begins
Testing Phase

Attempt to determine parts of the code that are incorrect

Good design structure makes testing easy

A good test case should detect an undiscovered error, not show that a program works correctly

Independent Testing

Alpha Test
Beta Test
Software Requirements Document

Title Page

- Project Name
- Author’s Name
- Date

Table of Contents

Introduction

- English description of what the project should do
- Not too much detail

System-Related Requirements

- Brief description of the minimum hardware and software related requirements (this should include a description of the server-related facilities (both hardware and software), and client-related facilities (both hardware and software)

Software Requirements

- Requirements that state the actual operations that the system must implement.
- Each operation should be addressed individually.
- "Sectionalize" requirements: Break each requirement down into single specifications, numbering each item in an outline format. Do not produce long paragraphs of description without itemizing them.
- Include screen diagrams to help user visualize interface.
Outline Format for SRD

**Typical non-outline form of requirement specification:**

For each question, they should be recognized by their number. The user should be able to mark their selection on the screen. The user should be able to change their answer at any time, until they press the submit button. The survey administrator can control the background color on the webpage, the background image displayed on the webpage (if specified), the title on the webpage, the number of survey questions, the text-content of each survey question and the choice of answers, the minimum and maximum number of selects that can be made, provide protection against the user choosing more than the maximum number of selections, or less than the number of selections, and to control when survey questions will appear on new webpages.

**Same specification using proper outline format:**

A. **General Question Information**

1. For each question, they should be recognized by their number.
2. The user should be able to mark their selection on the screen.
3. The user should be able to change their answer at any time, until they press the submit button.
4. The survey administrator can control:
   a. the background color on the webpage,
   b. the background image displayed on the webpage (if specified),
   c. the title on the webpage,
   d. the number of survey questions,
   e. the text-content of each survey question and the choice of answers,
   f. the minimum and maximum number of selects that can be made,
   g. provide protection against the user choosing more than the maximum number of selections, or less than the number of selections,
   h. and to control when survey questions will appear on new webpages.
Design Document

Title Page
- Project Name
- Author's Name
- Date

Data Structures
- Major data structures named with descriptions

Hierarchical Structure

```
Project
  Initialize
  Process
  Terminate
    Get Input
    Process Data
```

Pseudocode
- Major blocks denoted in Hierarchy Chart
- Code not necessarily related to a specific programming language (pseudocode should be a higher-level thought process as opposed to actual code)

Examples of Design Documents written by other 4010 students are available in PDF format on the class website.
Pseudocode Example

Note: Values inside (parentheses) represent subroutines that are expanded later in this pseudocode

main [in pool.cpp]
  if script not called from the web, then
    display identification message and exit
  if script called with "GET" method, then
    if there was no parameter (indicating which pool to process), then
      display error message and terminate
    (get ini-info for specified pool) from parameter
    (get data file info for this pool)
    (display introductory webpage info)
    (display main webpage showing pool pot)
  else if script called with "POST" method, then
    setup forms processing
    get the pool name from forms field
    if field not present, then
      display error message and terminate
    (get ini-info from specified pool)
    (get data file info for this pool)
    (display introductory webpage info)
    if "Submit Selections" button was pressed, then
      (submit-selections)
    else if "Display Charges" button was pressed, then
      look at each player and determine which squares he's selected
      and total his charges, displaying the results
    else if "Redisplay Pool" button was pressed, then
      (display main webpage showing pool table of values)
      else unknown request, so display error message and terminate

  else display "Unknown reference method" and terminate
  terminate

=========================================================================

display introductory webpage info
  display initial HTML heading information, using the color/image
  information specified in the INI data
  display the Info lines, if any

=========================================================================

display main webpage showing pool pot [in display.cpp]
  create form info
  display player selection
  (display board)